

Issue 37 | May 2012

# blender art

MAGAZINE

Blender learning made easy...

## *Cool Tools*

Video editing with Blender

How to construct and use a chrome light probe key-chain

Evaluation of Acoustics using Ray-tracing

The Art behind the Ragdoll

A Pawn Still in a Hurry... 11 years later

**EDITOR** - Gaurav Nawani  
**MANAGER/EDITOR** - Sandra Gilbert  
**WEBSITE** - Nam Pham  
**DESIGN** - Gaurav Nawani

**PROOFERS**  
 Brian C. Treacy  
 Bruce Westfall  
 Daniel Hand  
 Daniel Mate  
 Henriël Veldtmann  
 Joshua Leung  
 Joshua Scotton  
 Kevin Braun  
 Mark Warren  
 Noah Summers  
 Patrick ODonnell  
 Phillip  
 Ronan Posnic  
 Scott Hill  
 Wade Bick  
 Valérie Hambert

**WRITERS**  
 Frans van Dunné  
 Oscar Baechler  
 Thomas Krijnen  
 Thaleia Deniozou  
 Arnar Gauti Ingason  
 Dreamsgate  
 Alban Denoyel

**COVER ART**  
 Factory Lot - by Tiago Nunes

<b>Evaluation of Acoustics using Ray-tracing</b>	<b>6</b>
<b>How to construct and use a chrome light probe key-chain</b>	<b>10</b>
<b>Evaluation of Acoustics using Ray-tracing</b>	<b>12</b>
<b>The Art behind the Ragdoll</b>	<b>15</b>
<b>A Blade Phone</b>	<b>19</b>
<b>A Pawn Still in a Hurry... 11 years later</b>	<b>21</b>
<b>Sketchfab</b>	<b>23</b>

## Upcomming Issue 38 - "Power of Compositing"

Looking for tutorials or "making of" articles on:

- How to use the Compositor to create amazing images/animations
- Special effects
- Using The Motion Tracker
- Using the Compositor with Cycles
- How you use the compositor in your work flows/ projects



**Sandra Gilbert**

Manager/Editor

*"it dawned on me that in a world where everyone walks around with a cellphone that more often than not does far more than make just phone calls"*

When I picked the theme for this issue I had a fairly specific thought in mind. I was thinking: Blender features and tools and real world tools. For real world tools I had in mind actual tools that you would use to accomplish a specific task, like a hammer or screwdriver, but of course it could cover just about any real world tool. I was imagining specialized tools for, I don't know, dentistry maybe or something used in a lab. It wasn't until Arnar contacted me and asked me if I considered a phone a tool that I had to stop and rethink my definition of "Tool".

As I sat and considered it, it dawned on me that in a world where everyone walks around with a cellphone that more often

than not does far more than make phone calls, that yes, phones are definitely tools. Which of course led me to consider all the things that we use on a daily basis that could be considered tools. I personally have an Android based media player that I carry everywhere and you would be surprised at the things I do with it on a daily basis. Then again, you probably wouldn't be that surprised at all. These days mobile devices make our lives easier. We can track and take care of so many things with them that most of us would be severely put out if someone took them away.

So equipped with a new definition for tools as being anything that makes a task or our lives easier, I thought about Blender. There are many features of Blender that definitely make tasks easier, and of course some are more obvious than others. But there are those tools that we use constantly and often don't even think of them as tools or cool. I know from watching Jonathan Williamson's tutorials that he makes heavy use of the grease pencil to illustrate what

he is talking about. David Ward flips in and out of sculpt mode to use the grab and smooth brushes to even out his models.

My current personal favorite tool is Cycles. And not for the popular reasons of obtaining a realistic render or the ease of setting up materials and lights. I admit those are cool, but I like Cycles because it gives me immediate feedback on whatever I just did or adjusted on my model or in my scene. I do a lot of poking and testing of materials and lights, so the instant or semi-instant feedback is awesome. I can set the samples fairly low and still get a good idea of how it will look. This definitely makes things flow easier for me.

So what is your favorite tool or feature, what is your "go to" tool? Are they just cool for coolness sake or do they make life easier? No matter the answer, we have a bunch of cool articles for you to enjoy and add to your "making life easier toolbox".

Let's get started!

# Izzy Speaks: Unsung Awesome Feature



## Izzy Speaks

From humble beginnings as some in house software, Blender has grown into a powerful 3D tool with an ever growing number of cool features and tools sure to please any 3D enthusiast. The latest round of tools to be added has of course set off a serious flurry of testing and experimentation to see how things work and what can now be done.

I often think that this part of the development cycle is the most fun. Everybody is playing and testing, pushing the limits of what can be done and doing their best to “break it” so that bugs can be found and squashed. It is a highly creative time in the Blender community. New images and test scenes seem to appear by the minute as everyone gets comfortable with the new features. Soon after, tutorials start popping up to explain the finer details for those of us that are running into problems.

Most everyone would agree that Blender has some amazing tools and features, but one awesome feature of Blender often goes unrecognized, and that is the dedication of the Blender community to help everyone use Blender to its fullest. It is a rare thing indeed to be able to email or message just about any user and get help with a problem. We might not always know all

the answers, but most of us are very willing to help others track down the information.

Apart from the community-wide willingness to help, there are members of our community that go above and beyond to leap into the learning gap as quickly as possible to help the rest of us out.

I think it is high time we gave them a huge high-five for the tireless hours they put into creating quality tutorials and Blender Education that makes the learning curve so much easier for the rest of us.

The following is a list of my favorite tutorial spots.

### BLENDER COOKIE Blender 3D Tutorials and Training

**Blender Cookie:** run by Jonathan Williamson and the rest of the Cookie Crew.

Jonathan has an almost magical ability to post relevant tutorials just as features are being added. Jonathon has explained that this is only possible because he downloads new builds daily and tests features as they are being implemented. This gives him just enough of a head start to be ready to

help the rest of us.

Often for me, the best part of Jonathan's tutorials is not the actual topic he is discussing (which of course is valuable), but all the tips and explanations that he includes as he works. I especially appreciate all his 'little talks' about topology and clean modeling techniques.

In addition to Blender tutorials, the Cookie network also offers tutorials on **Concept Art** and the **Unity Game Engine**, often a large part of many Blender users' work flow and projects. The addition of these sister sites allow Blender users to expand their project parameters and learn new things that touch upon their Blender projects.

The majority of tutorials on Blender Cookie are free for anyone to watch, so while it is not necessary to be a **Citizen** to enjoy what the Cookie Network has to offer, in my opinion it is well worth the very reasonable price that they charge.



**Blender Guru:** run by Andrew Price

Andrew produces high quality tutorials that provide a well rounded look at



# Izzy Speaks: Unsung Awesome Feature

projects, generally from start (modeling) to finish (compositing). For those interested in “product shots”, Andrew has done several excellent tutorials on how to show an object to best advantage. These are some of my favorites from Andrew's site.

He has also started a well received “**Nature Academy**” where he teaches users how to create realistic nature scenes.


 The logo for BlenderDiplom, featuring the word "Blender" in red and "Diplom" in black, with a red horizontal line underneath.

**BlenderDiplom:** run by Gottfried Hofmann

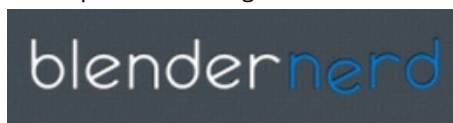
Tutorials on Gottfried's site often focus on special effects and cool new ways to use Blender features and tools. There are also a variety of extremely interesting projects often presented as a series as well as interviews with members of the Blender community.



**blendtuts:** run by Oliver Villar

Oliver offers a nice variety of tutorials in both English and Spanish. In addition

to all the tutorials on his site he also has a Hologram Project for sale in his e-shop that covers concept, modeling, materials, animation, rig, compositing and lighting. The project is available in both Spanish and English.



**Blender Nerd:** run by Greg Zaal, Rex Harby and Tanner Casey

They have recently launched a brand new look to their site, which had already been full of great tutorials on a variety of Blender topics. One interesting new addition to their site is the ability for users to upload their own tutorials to share with Blender Nerd users.

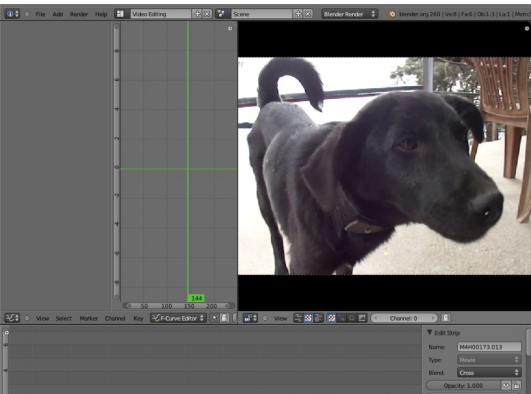
There are of course many more valuable tutorials sites in the Blender Community, and of course with that many to track it starts to become difficult to keep up with who is doing what. Which is why I cheat. I discovered that by following one very connected Blenderhead, I could be kept informed about new tutorials, news, images and cool projects.

**Terry Wallwork**, better known as **Adventures In Blender** has an uncanny ability to always know what is going on where in the Blender

community. I am beginning to suspect that he has found a way to hard wire himself into the internet, which while possibly odd, works out well for me. I follow him on **Twitter** and **G+**. Terry also has an **Adventures in Blender** website where he reviews Blender books and tutorials series. His interviews are always well written and very insightful.

Let's all give our Blender educators a big Blender hug and do what we can to support their continued efforts on our behalf ●

# 3D WORKSHOP : Video editing with Blender



by - Frans van Dunné

## Introduction

Blender has grown into an amazing tool, standing tall at version 2.62, and for those who did not have to unlearn the old interface, the new interface is much easier to get accustomed to than it was in the version prior to 2.5.

Although I have followed Blender and played with Blender for a number of years now, I have only now moved to the Blender Video Sequence Editor (VSE) for production. The reasons were twofold, on the one hand I got stuck in iMovie when I tried to sync my audio tracks precisely to the video. And on the other hand my main PC now runs Ubuntu Linux, and no longer OSX.

iMovie is amazing for doing a fast, no hassle edit of a video. Adding a recording or a soundtrack is just as easy. But it only goes so far. Blender on the other hand allows you to do almost anything you can think of, especially in combination with FFMPEG. But expect to be tearing out your hair at times figuring out how.

This is by no means intended as a manual, or even as a description of

how to do things in Blender. Take it more as a long story about my experiences moving to this platform. The Blender manual itself is good, but you will need to rely on the chapters for version 2.5, which do apply but do not answer all the questions.

## Setting up additional software for audio

The hardest part I found was managing audio. Although the Blender installer comes with FFMPEG built-in, it will not always render the video with the audio that you have included.

The Blender Output and Encoding interface allows you to set all kinds of combinations of options that simply do not work, especially when you try to add an audio track. This only stops from being annoying when you simply separate the audio workflow from the video workflow completely.

The set of tools I ended up with are:

## For Video

- Blender
- FFMPEG (Make sure you install version 8.0 or higher!)
- xjadeo
- Gimp

## For Audio

- QJackl (Jack)
- Ardour + LAPDSA Plugins
- Audacity

If you are using Windows or OSX, the list would be shorter for audio: you can use Reaper now that it has the ability to add a video track.

Why so many programs? Well, what you give up in ease of you, you gain back in having control over every detailed aspect of the video creating process.

To give some context of what I

do, I will describe where the different components of a recent product presentation movie I made.



## Video:

- JPG AVI Recorded video with Nikon D90
- Video stills created in part in Microsoft PowerPoint (exported as high resolution PNG) and edited in GIMP.

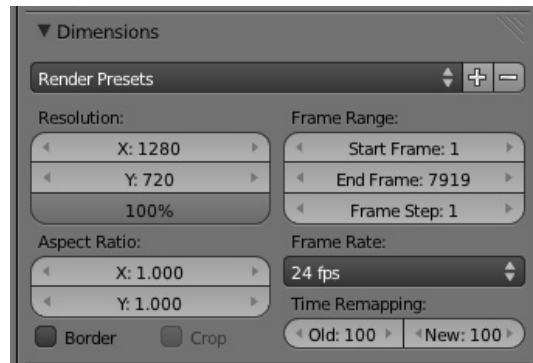
## 3D WORKSHOP : Video editing with Blender

### Audio:

- Music soundtrack
- Recorded voiceovers

### Setting up and importing video

Make sure you set up the framerate of your video file in the Dimensions tab before you start working. For instance, if you are working with Nikon D90 you set the frames per second to 24 and the frame size to 1280 x 720. When you import your movie clips to the VSE it will set the frame rate according to these settings, not according to the video you are importing. So if you do not do it before you start, you would have to import all your scenes again.



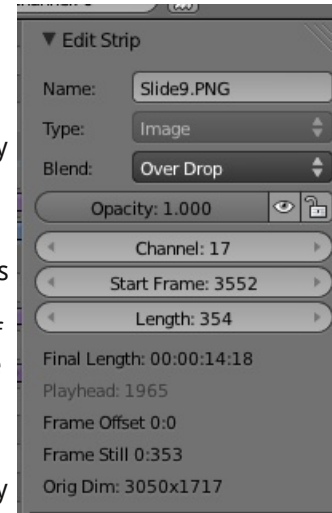
Some video tracks I had converted earlier to the Apple intermediate codec, and I no longer had access to the AVI originals. These need to be exported using Apple Quicktime Pro, and using H.264 multipass encoding worked for me after some experimenting. In my case the audio of the video footage was never used, so I paid no attention to that.

Next I imported all the selected video tracks into blender, using separate channels for the video and the still images. The still images need to be on the higher number channels and have a transparent alpha channel. You can then set to "drop over" so that your video will appear below. Scaling down the images to 1280 x 720 works, however, I sometimes run into trouble with the de-interlacing settings. Just one odd pixel in your greys can lead to a white or off-colour pixel in the render in Blender.

I have not figured out exactly how and why, but the only solution I have found is to increase the amount of pixels (the resolution) of the images. If you do not have a higher resolution available, try removing (pretty much hit and miss) one of the lowest level transparent greys in Gimp. If you hit the right one, then suddenly many if not all of the white pixels in your render disappear.

### Adding stills

There is a way to use Blender for moving titles



and rolling credits, but I have not figured out yet how to do that. Instead I used a PowerPoint presentation as the basis for all the titles, subtitling, and credits, and edited these in GIMP. For the subtitles instead of using X exports from PowerPoint, I created one layer with the transparent subtitle pictures, and then X layers with the text. If you export from Gimp with the setting "export visible layers" you can quickly export each text as a separate frame. And it relieves you from going back and forth to PowerPoint and add the transparency to every single frame.

For adding the transparency, the option to substitute a colour (white in my case) for an alpha channel is great. For areas that where this is not wanted (a grey logo for instance), just use the selection tool in blender, and select the areas where white can be substituted safely before making the substitution.

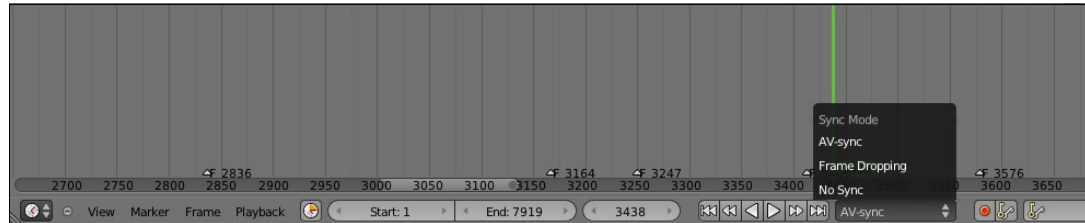
### Syncing audio

Either your soundtrack is ready and you only need to sync the video to it, or the soundtrack will need to be edited in iterations to fit your movie. The latter happens for instance when you are preparing a presentation with spoken text.

In either case, when you start syncing the audio to the video, make sure you have AV-Sync on. It will tell blender to always follow the audio dropping frames if it has to. One caveat: if you have included high resolution

## 3D WORKSHOP : Video editing with Blender

pictures stills, Blender is not able to drop the frames, and your sync to audio will be momentarily off. One way is to include a low resolution picture while editing and replacing it in your pictures folder with the high resolution version just before rendering. The



other is to trust the time marker you set based on the audio wave-form and then check in the render if the timing is correct.

Next load the soundtrack that you have (I use Channel 0 as it is my reference for the placement of the rest) and make sure you set the option to Draw Waveform. If you have snippets of sound (like blocks of spoken text from your movie or presentation script), then you can add them one by one or drag them from the Blender File Browser.

You can do the same with the audio-snippets of your voice-over draft. When editing this way I put more emphasis on getting the length more or less right based on these voice-over snippets. I then make a final version of the voice based on that, and a final edit of the video after importing the complete voiceover render (see below).

### Rendering your Audio Track

You will need a DAW like Ardour on Ubuntu or Reaper on windows to make sure you can manage the fade-ins and fadeouts of music and voice if you use a soundtrack. What I did was to make a leader track of the voice with Ardour and chop it up in snippets with

Audacity (Audacity is great for quick and dirty edits like that). These snippets then went into the Blender VSE. With these leaders I could set the length of the videos and stills, and then decide on whether and how to redo the voiceover.

For the final edit, I moved the audio-less render of the movie to a file that can be read by xjadeo (or reaper if you work on Windows/OSX). You do not need a high quality version of the video, just enough to see all the frames, and make sure that you have enough processing power to edit the audio.

One way is to reduce the weight of your render is to reduce the frame size by half, in this example export the 1080x720 movie to a 540x360 size.

All transport in Ubuntu is best handled by Jack, so you need to adjust Ardour accordingly and you do need to set the frame rate for the Ardour clock (under options -> FPS). My

biggest challenge has been the fact that there are only 24 frames in a second. And that does not always allow for precise timing. There is no way around that in a long audio track, the 24 frames are fixed. However, voice-overs can be cut and readjusted more precisely.

After your final edits of the soundtrack, import it back again into your Blender project VSE. Make sure that you never change the start point of the audio track, or you will have to transpose all your frames to match the start. The end can be different, perhaps you have created a fade out where you want the movie to end. Select the new audio track and copy the length to the total length of your render.

This needs to be done, because to combine the audio and the video track in FFMPEG both the sound and the video file need to have the exact same length. There is an -shortest flag that tells FFMPEG to use the shortest of the two files as the length for the output. But then you lose the control that you went to all this length for on how the movie ends.

### Rendering

Once all this is done, you will want to render the movie to a lossless format. It gives you a base-file that will serve as input for whatever file format you need the movie in (and these can be many – and once it is rendered you do not want to render it again). The more processors you have in your GPU the better.

Even though for whatever reason Blender

## 3D WORKSHOP : Video editing with Blender

does not seem to make use of all the cores available, FFMPEG most definitely will at the lossless settings.

The settings for the current version of FFMPEG to encode the audio.

```
ffmpeg -analyzeduration 1000000000
-i
../AudioFiles/myFinalAudioRender.flac -i myFinalVideoRender.avi -ab 320k
-ar 44100 -vcodec libx264 -preset
veryslow -tune zerolatency -threads
0 myFinalMovieWithSound.avi
```

The key flag here is `-tune zerolatency` that will improve the quality of the audio syncing to your movie. Then it is a matter finding the FFMPEG to transcode the lossless version of your movie to the movie format you need.

### Closing note

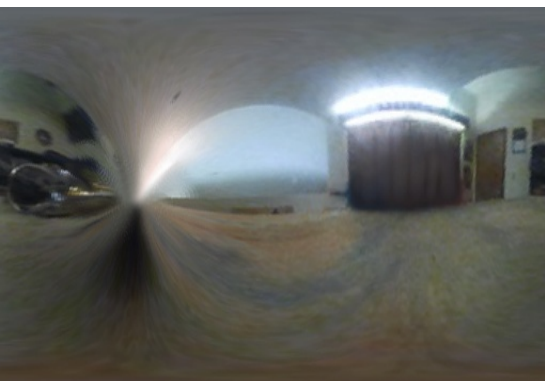
It took me the best part of two weeks to get adjusted to the new workflow. But I would not trade back for anything. The best part is the Blender interface, which allows you to define exactly what you see on your screen. While editing I have a closeup of the wave form of my audio file in the left hand corner for instance, where the playhead (the green bar indicating what the current frame) will be visible across both the video sequences and the close-up of the audio.

I also realise that there is much, much to learn. In the above I have not even mentioned using compositing nodes to manipulate your video tracks. Or the possibilities to combine

video with rendered animations. I do find that learning those aspects of Blender take time. As I said at the start, if it is a quick edit you want with nice out of the box features for adding text etc. do not bother with Blender, but stay with what you are comfortable with right now. But if you want to expand your tool-set in a creative and challenging environment, then by all means, Blender is the way to go ●



# 3D WORKSHOP : How to construct and use a chrome light probe key-chain



by - **Oscar Baechler**

## Introduction

I've been reading up on HDR lighting for 3D. I'm no photography buff, but it's a handy way to light 3D scenes without doing a lot of work. Then spring came around and during walks around Seattle, I'd find myself thinking, "What a beautiful day! If only I had a mirrored ball and an expensive camera handy."

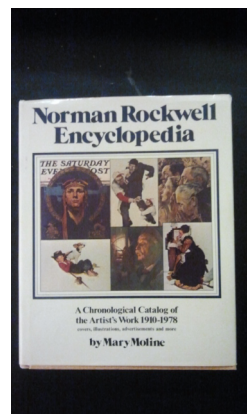
Then again, I usually don't need high res hyper-RAW pics for scenes. A relatively low-res image gets you pretty far. I've always got a quick and dirty camera via my phone. But what about a quick and dirty light probe? Hence, a crazy scheme is born: to make a chrome ball key-chain, and photograph it wherever I happen to be.



First, the chrome balls. They can be found in all sorts of shapes, sizes, and

quantities on Amazon, but you'll probably have to buy in a pack of 10 or 20. Why not assemble a Blender User Group, like Seabug (this Saturday!) and share the leftovers with others?

Shopping around revealed some bad news. Handier folk than I recommended either welding or tapping a screw to fix this to a key chain, but I lack both the skill and equipment to do so. Additionally, paying a professional to do one of these options would cost far more than the metal balls themselves! (quotes ranged from \$60 to \$600.) Hence, we turn to the glue option.



We'll need a good work surface we don't mind getting beat up or glued. I used this Norman Rockwell book, because like Picasso and the Ninja Turtles, he's the sort of zeitgeist-hogging artist that even the lowliest pleb can namedrop. Hence, his literature over-saturates the pseudo-bourgeois coffee tables of Americana, and any number of volumes can be

obtained for a tuppence at your local Goodwill.

When shopping for glue, search for ones that specify metal-on-metal contact, like this Loctite epoxy. Additional tools include a metal file for roughening surfaces before gluing and removing glue later, a magnet for easy anti-roll propping (old CDs work well too), disposable gloves to avoid emergency room visits, an eyelet screw, and a thicker washer for wider base and increased stability.



File the surface of the contact point for each chrome ball as well as the washer/screw contact point. This roughening will help the glue bond.

Glue the two things together, and also try to get some glue inside the washer along the screw thread. Hold them upright, with the magnet helping you out, but be careful not to let the eyelet

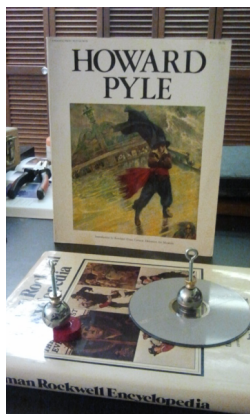


## 3D WORKSHOP : How to construct and use a chrome light probe key-chain

droop; it'll zoom right to the magnet. After a few minutes, it should be dry enough to stand on its own, but is still structurally weak.

I switched one ball to a stack of CDs for the remainder of the drying. While you're waiting, why not read up on a grander and more obscure artist for future name dropping? Ah, Howard Pyle. That's the ticket! Master of color, the best early illustrator to make it big mass marketing to the lowest common denominator, and an educator to whom many great students trace their lineage. Why, imagine how he'd feel knowing now that his children's book fancies are now prized among...good news! The glue is dry.

Ready to have your mind blown? When you look at a reflective ball, you see more than the 180 degrees facing you! This means that when you photograph your ball, you'll have to edit out two things:



yourself and your keys. To minimize other artifacts that will clog your photo, use the file to remove excess glue.

Learn what you can about operating your crappy camera phone. Try to get good resolution. Also, although it illustrates the hardware, try to position the your keys behind the ball, rather than on the side.



My camera crops with only a little bit of uprezzing to put this exactly in the center of a 512x512 texture.

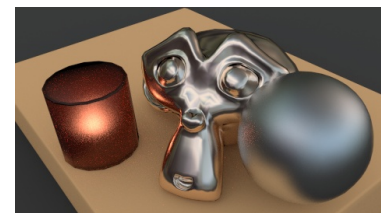


I distort this to a rectilinear map using the Flexify 2 filter for Photoshop from Flaming Pear. Although I haven't tried it, there's also Panorama Tools for GIMP, and probably some other stuff. I dunno, google it. The settings I used were Mirror Ball to Equirectangular, which jams it into 256x512.

I cleaned out two artifacts: myself in front of the closet, and the keys along the left pole. In retrospect, I should have cleaned out the ball's contact shadow (AKA ambient occlusion for you 3D types) before remapping as well.

Now to get this in 3D! I used two setups, Cycles and Internal for Blender. The Cycles render is simple. On the World coordinates tab, change the Color to Environment Texture, and select your equirectangular texture, leaving the Vector at Default. Turn on Ambient Occlusion, and under Settings turn on Sample as Lamp.

Hey hey, instantly lit scene!



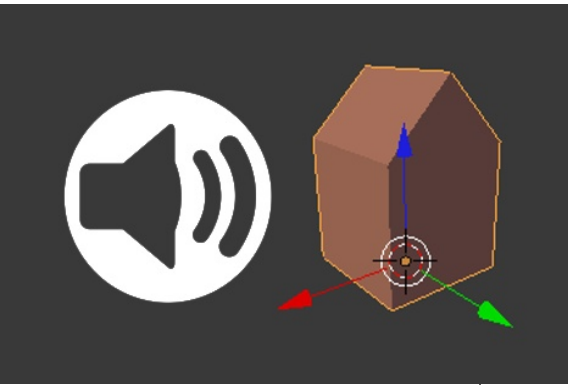
For the internal renderer, you have to do a lot more finagling. First, the World should be mapped to Real Sky. Turn on AO, Environment lighting (set to Sky Texture), and set Gather to Raytrace, and enable raytracing in the Render tab.

For the World texture, instead use an Image type texture (instead of an Environment Map) because this is quick and dirty, and thus can't use Env Map's fancy goodness. Select your equirectangular texture, and map to Vector and Default.



Feel free to play with my test file ●

# 3D WORKSHOP : Evaluation of Acoustics using Ray-tracing



by - **Thomas Krijnen**

## Introduction

I am a big fan of ray-tracing, and who in the right mind would blame me for that? In fact, I am such a big fan of ray-tracing that I have applied it to something besides the visual domain, namely to sound. Now this may not be the first tool that uses ray-tracing for analysing the acoustics of a

digital scene, and by all means not the last either.

But in this article I will introduce a free open source auditory ray-tracing tool for Blender, called E.A.R: Evaluation of Acoustics using Ray-tracing.

To answer the question why I would create such a tool, let me briefly introduce myself. I am an architecture student with quite an unhealthy appetite for computer technology. Somewhere in time I felt that as an architect, I was overloaded with the visual experience in my buildings.

So I started to work on an easy-to-use artistic tool to give architects a way to evaluate the aural experience of their buildings. The tool soon evolved into something that I think is generally

useful for everyone who has an interest in computer graphics, film making and special effects.

## E.A.R.

E.A.R. enables modelers and designers to actually hear what spaces they conceive will sound like. Reflections, volume, material properties and movements are all taken into account to come to an overall auditory representation of the scene. Using E.A.R. is quite easy.

Normally a 3D scene is created with light sources and geometry that reflects light to the camera. With E.A.R. sound sources are placed and geometry reflects the sound to a listener. In short, that is all there is to it. However, as it goes for everything we get for free in life, in order to get good results, it is generally nice to have some understanding of how things work under the hood.

## Technical background

Digital sound files consist of a lot of samples per second. Tracing all these samples individually through space would simply take too much time. That is why E.A.R. uses ray-tracing to generate an Impulse Response for each sound source, independently of what sounds come out of it. This

Impulse Response describes the decay of sonic energy from the sound source at the location of the listener. Because sound behaves differently in respect to different frequencies, Impulse Responses are created for three frequency ranges.

People move, as do sound sources. To incorporate these movements in the simulation process, the movement is divided into key-frames. For every key-frame an Impulse Response is generated. These Impulse Responses are interpolated as the movement from one key-frame to the next progresses.

Another thing that is different from visual ray-tracing: when people move they do not generally begin to emit light, however, in our movements we do generally create sounds, for example by our footsteps.

This is the reason E.A.R. comes with an easy way to generate these sounds from an animated path of a listener. When all the Impulse Responses are generated, the sounds, which are emitted by the sources, are processed to incorporate the rendered acoustical response of the room. This is a step which is called convolution.

To let you in on a little secret, from a purely physical stance, using ray-tracing to simulate the propagation of



# 3D WORKSHOP : Evaluation of Acoustics using Ray-tracing

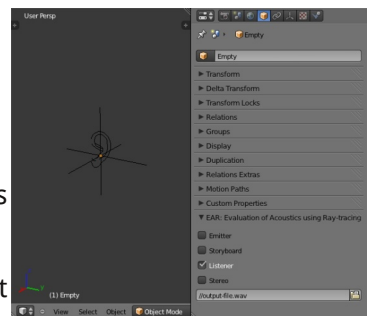
sound is not entirely correct. Sound behaves much more like a wave than like a ray. Because of the large wavelengths of sound, phenomena like diffraction can be very prominent. These phenomena are hard to reproduce exactly using ray-tracing. Nevertheless E.A.R. can give a valuable artistic impression of how a space sounds.

## Now quickly back to Blender

Good, you are still with me. Let us quickly take a peek at how this all works in Blender. E.A.R. comes as an add-on that you have to enable. By doing so, some extra panels show up in the user interface. These allow the configuration of some basic settings and specify what objects behave as sound sources, what sounds come out of them, what objects reflect sound and what objects are listeners. This article is not going to be an exhaustive user manual, for I will not bother you with every little detail. I will just show how the key concepts relate to the Blender interface.

## Listeners

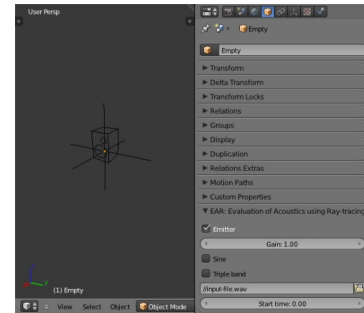
Listeners are created as Blender Empty objects. Each listener has its own output path for the sound file that



enables us to play back how the scene sounds at its location.

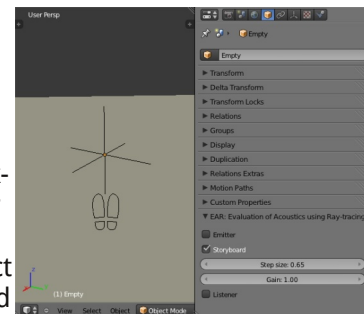
## Sound sources

Sound sources can either be an Empty or a Mesh. In the latter case sound is emitted from the faces instead of from a single point. There are several ways to define the input of such a sound source. The simplest way is to point it to a sound clip on your hard drive.



## Storyboards

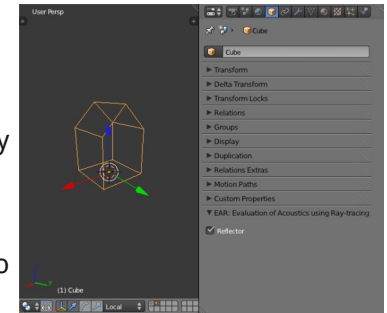
Storyboards are objects intended to help in the creation of auditory walk-throughs. For each step such an object takes, a sound is selected based on the material that is being walked upon. Furthermore, when such an object moves through a Portal object, which for



example could represent a door, a corresponding sound can be played as well.

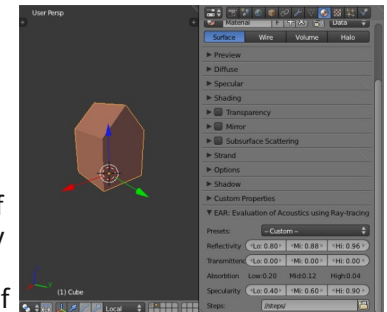
## Geometry

In order for meshes to reflect sound they have to be specifically told to do so in the interface. There are two reasons one should not create too detailed a geometry. First, the ray-triangle intersection code is not that fast. Second, and more importantly, because of its large wavelength and diffraction, sound would in fact move around those small details anyway.



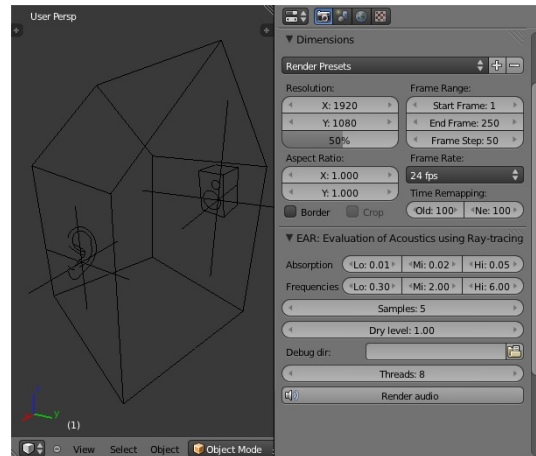
## Materials

Similar to visual ray-tracing, materials define how the rays bounce off of the geometry in the scene. When a ray of sound hits a mesh face the sonic energy is partly



## 3D WORKSHOP : Evaluation of Acoustics using Ray-tracing

absorbed, partly reflected and partly transmitted. Materials define the ratio between these three characteristics. The fourth property of a material, called Specularity, defines to what extent materials reflect and refract rays in diffuse or in a mirror-like way. There are also some material presets that can be chosen from. For use in conjunction with Storyboard objects, a set of sound files can be specified for materials that dictate how the material sounds when walked upon.



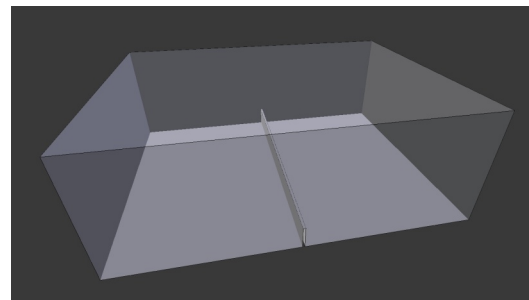
### Scene settings

Some general settings can be applied as well. Absorption defines the percentage of sonic energy that gets absorbed by one meter of air for the three respective frequency ranges. Increasing the number of Samples will be more accurate, but will also take more time. Using 5 or 6 will usually be fine. Key-frames are generated at equal intervals based on the

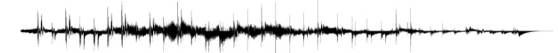
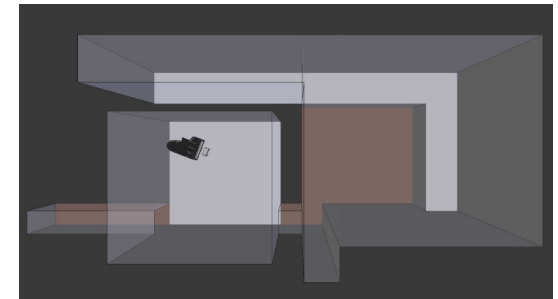
Frame step setting in the Render panel. This setting is only relevant if any of the objects are animated, but in that case, be sure to increase the value to around 50, based on the length of your animation, which is defined by the Start frame to End frame. The other settings are a tad bit more advanced, so I will not touch on those.

### Examples

Well, a sound says more than a thousand words, so with pleasure I provide you with two examples of what E.A.R. can do for you. In the first example no animation of sound sources and listeners is defined. It shows the dramatic effects one can create with a very large reverberant hall. One that makes a mere finger snap resonate for seconds. The second example is much more subtle and intended to highlight the possibilities of animating E.A.R. It shows a progression of spaces explored by a visitor against a backdrop of piano music.



Example - scene1.mp3



Example 2 - scene2.mp3

### Concluding remarks

Be sure to keep an eye on the explauration website: <http://www.explauration.org/>, it will feature any new developments around this Blender add-on and on the auditory experience of architecture in general. On this website you will also find a more detailed inspection of the two examples, as well as a downloads section and hands-on instructions to get started. E.A.R. is open source software; the code is available on github ●



# MAKING OF : The Art behind the Ragdoll



## Introduction

The 'Ragdoll' is an interactive art game developed using the Blender Game Engine as part of the Design and Digital Media Studio Project (DMSP) module at Edinburgh College of Art,

University of Edinburgh [Fig.1]. The DMSP course, organized by Martin Parker, is a unique opportunity for highly creative students from various design disciplines to work collaboratively in a studio setting, towards the implementation of experimental design work for public display. Students are assigned together in small groups working under supervision on different project briefs, ranging from digital installations and real-time performances to 3D animations and interactive games. This article explores the journey of the 'Ragdoll' game from concept to implementation and discusses its nature as a self-proclaimed art game.

The idea behind the project was based on art games as an innovative digital medium for creative experimentation via real-time, interactive 3D environments.

Inspirations constituted games like 'The Path', an atmospheric, beautifully designed art game, the monochromatic, eerie 'Limbo' and the platformer 'Braid', an allegorical journey based on time and regret. Looking at such inspirational existing games and being informed by the interests of the project supervisor and the creative vision of the participating students a dark-themed storyline was adopted. Film noir, horror and fantasy were the three main design elements that would characterize the atmosphere of the game.

Emphasis was put on the artistic merit of the design, while the pursuit for a unique aesthetic quality quickly became the main objective of the team working on 'The Ragdoll'.

*"From my point of view art games are characterized by an extraordinary, unconventional look, standing out for their aesthetic beauty and imaginative design, and for us 'The Ragdoll' was no exception. We wanted to make a game that could evoke some kind of emotion; an immersive memorable experience"*

Thaleia Deniozou, Project Supervisor

During the initial discussions the decision to build an exploration game was made. The main aim of the team was to treat the game that was about to be designed as a piece of

interactive 'art', in the sense that the focus was on the game's potential for meaning; thus the storyline, the narrative behind the exploration, emerged as particularly important. 'The Ragdoll' would be an imaginative world, inside which the player could explore, live the story and have an immersive game-play experience without being distracted by challenges often appearing in conventional games like battles, races, skill development and competitive goals. The story as well as the aesthetic quality of the design, were top priorities.

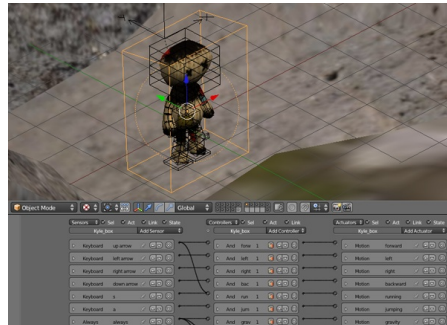


[Fig. 1] The Ragdoll Game

The story that drives the game-play of 'The Ragdoll' evolves around the quest of the main character, an old doll named Kyle [Fig.2], who wakes up in a strange, unfamiliar place he has never seen before. The Valley of Forgotten Dreams, as he later figures out is the name of this weird place, is an enchanted, spooky forest. Lost and confused, Kyle finds an ally in an old magical tree who offers to help him

by - Thaleia Deniozou

# MAKING OF : The Art behind the Ragdoll



[Fig.2] Kyle the Ragdoll in Blender GE

return back home if he completes three tasks. Kyle needs to bring back the Goblet of Eternal Water to water the tree, the Golden Harlequin Beetle to fertilize the tree and the Mighty Dagger of Thousand Thunders to cut the death twigs and revive the old tree. Each one of those tokens is situated at the end of a dangerous path protected by mighty poisonous plants. Left without a choice the ragdoll agrees and the game begins.

In terms of mechanics the game had to be kept simple as balance needed to be achieved between building an engaging game experience but with a non-conventional game purpose in mind. The final product needed to be an experimental piece of interactive design ideally targeting various types of players. The game was not to be developed only with the gamers in mind, but addressing a wider audience and an unconventional

purpose, keeping in mind it could be played as a stand-alone game but that it could also act as an interactive piece suitable for Gallery display. Therefore interaction had to be kept simple and an amount of abstraction was necessary. In the end the team decided that simple mechanics coupled with imaginative design and an engaging back story was the way to go.

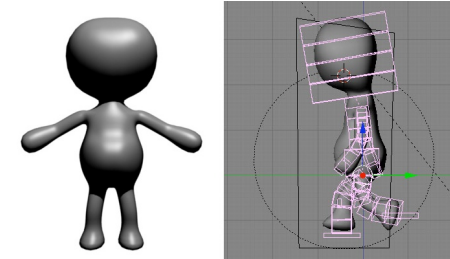
After the concept and back story were defined the group began modeling using Blender. The character was developed starting from a simple cube where a mirror modifier was applied and then was built up using extrusion and face modification to achieve the shape of the ragdoll. Then the model's faces were unwrapped and imported into Photoshop as a 2D image, where the texture was painted. A wool cloth texture with colored patches was used to emphasize the ragdoll aesthetic [Fig.3].



[Fig.3] Textured model

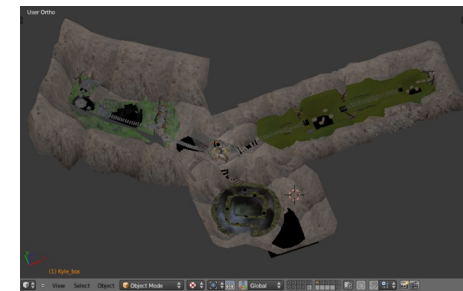
The next step was to rig the character. The process involved the application of bones in the ragdoll's body and

then the animation of the bones for realistic character movement [Fig.4]. The animation actions included a walk cycle, running and jumping states and a breathing state for the doll's standing position.



[Fig.4] Basic and rigged model

The next step was to model the game environment and props. As the aim was to provide a non-linear explorative game-play, the game world (Valley of Forgotten Dreams) was developed as three different dangerous paths at the end of which one of the tokens was situated [Fig.5]. Each of the paths featured challenges for the player to overcome, like



[Fig.5] The Game Environment. Three Paths

# MAKING OF : The Art behind the Ragdoll

poisonous plants, broken bridges and black holes the ragdoll should not fall in.

The three paths started from a middle area, where the tree was positioned and the ragdoll was waking up in, when the game started. From there three signs indicated the paths and the player could choose the order in which to follow them.

The first path was the 'Doll's Eyes' where poisonous Doll's Eyes plants existed throughout. At the end Kyle could find the Goblet of Eternal Water. The second path the 'Fly Amanita' was full of poisonous mushrooms able to cause hallucinogenic effects and Venus Flytraps that could capture and kill the ragdoll.

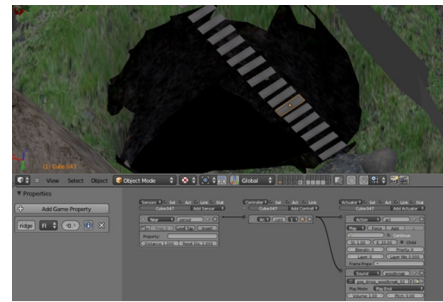
Successfully reaching the end of the path, Kyle could collect the Mighty Dagger of Thousand Thunders. Finally the third path, 'Poison Ivy' was named after the poisonous ivies located



[Fig.6] 'The Poison Ivy' Path

around it [Fig.6].

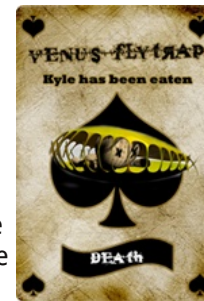
The main hurdle in this path was an unstable rope bridge Kyle had to cross and a spiky wheel of misfortune located over a deep ravine. At the end of the path the Golden Harlequin Beetle was waiting to be collected.



[Fig.7] An example of simple Logic (when character is near the bridge, animate it falling and play cracking sound)

After the modeling and texturing of the character and the environment were done, the team went on to develop the interactions, using Blender's Logic Bricks system [Fig.7]. Interactions were carefully organized to make navigation intuitive and the game easy to play.

Narrative cards as scene overlays were used throughout the



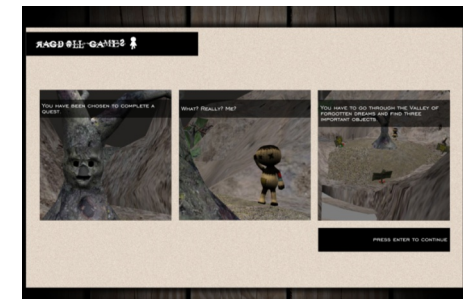
[Fig.8] Death Card

game play to indicate achievement when one of the three tokens was collected, or to signify death when the player fell in one of the traps, or was killed by a poisonous plant and the game restarted [Fig.8].

*"The team was quite creative with the narrative cards. They were well designed and served as a good mechanic to indicate state changes like achievements or failure"*

Thaleia Deniozou, Project Supervisor

Similarly story strips were also incorporated in the introduction of the game to narrate the back story and provide context to the player. These were displayed before the actual game-play began [Fig.9].



[Fig.9] Game start-up scene. Introducing the story

Sound was also one of the game's strong points as the team benefited from a Sound Designer and an Acoustics and Music Technologist. Game audio effects were recorded,

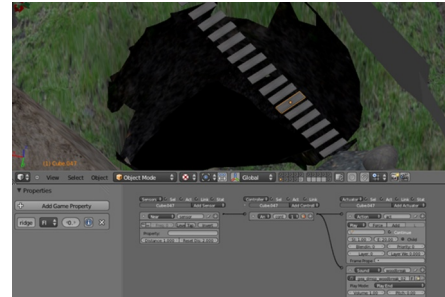
# MAKING OF : The Art behind the Ragdoll

then edited and finally incorporated into the game engine and controlled via logic. Sound effects were designed to be literal to the point of exaggeration and included, but were not limited to, the doll's footsteps, a small 'whoosh' sound when jumping, a flip sound when a narrative card was dealt or removed, a snapping sound when a Venus Flytrap closed, a humanistic yelp when Kyle died etc.

The background music of the game was designed to be immersive and informative, along the lines of adventure objective-based games.

The final game was presented at the University of Edinburgh in a public display in March 2011. During the presentation the team members gave a short talk on the game design process and their experiences working with the Blender Game Engine and then the audience had the opportunity to play-test the game that was installed and running on a number of computers.

The play-test was quite successful with the majority of participants commenting that they liked the graphics and the music and that they found the game's atmosphere interesting overall [Fig.10].



[Fig.10] Play-Testing during public presentation

*"I am very happy with the final result. When I suggested the 'Art Games' brief for the Digital Media Studio Project I had a specific idea of the final product I would like for the student team to produce in a three-month period, but my expectations were surpassed with 'The Ragdoll'. Claire, Heida, Lenka, Graeme and Hunter proved an amazing team of designers with a strong creative vision"*

Thaleia Deniozou, Project Supervisor

This year for the Digital Media Studio Project Thaleia Deniozou supervises two student groups working in a similar game design brief. Currently the teams are on the initial stages of development of two creative games entitled 'Edinbear' and 'The Great Escape' respectively.

*"Both games this year have great potential and I am happy to see that the teams are quite motivated and creative in their game design approaches so far."*

*The bar was set up high last year with 'The Ragdoll', but hopefully this year another set of great games will be produced"*

Thaleia Deniozou, Project Supervisor

## Students

Claire Meldrum, Heida Vigfusdottir, Lenka Bartosova, Graeme Arthur, Hunter Knight

## References

- The Path (Tale of Tales, 2009)
- Limbo (Playdead Studios, 2010)
- Braid (Number None Inc, 2009) ●



Thaleia is a digital designer and blender expert. She has been tutoring 3D design, animation and game development at the University of Edinburgh since 2009.



# MAKING OF : A Blade Phone



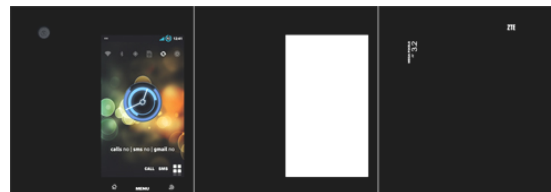
by - Arnar Gauti Ingason

## Introduction

My name is Arnar Gauti Ingason and I'm from Iceland. I recently started using Blender, but have been doing 3D work for some time. I have used Maya, 3Ds Max and

Cinema 4D but never actually owned a copy of them and wanted to use 3D software legally. So I checked out Blender and was very surprised by how much it had developed since I first tried it and rejected it (pre-2.5).

In the beginning of this project I sort of just looked around my desk for something to model, eventually I decided to do my phone since it had some interesting shapes and curves I thought might be a challenge to practice my modeling skills.



## Modeling

I started by searching for images of the phone online from straight angles

(front, back, sides) in a decent resolution. Also, I had the finished product right in front of me so I could view as much detail as I wanted.

The modeling technique used was box modeling but if I would do it again, I would probably use edge modeling since I think that would give me a better topology in the end. The most challenging parts were the camera and the top of the phone.

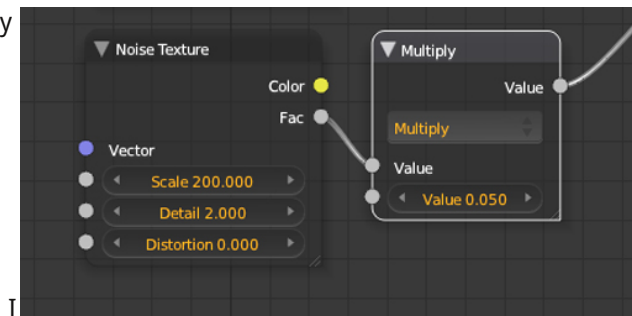
## Materials

For the glass, the screen is actually in the texture. I started by UV unwrapping the mesh that needed to have the glass material. In the UV/Image editor, if you go to UVs > Export UV layout you get an image with UVs on a separate layer. That image you can take in to Photoshop or GIMP and easily see where each part of the texture should go (in the finished texture you should delete the layer with the UV layout).

Since the the glass had to consist of several materials I made a simple color texture and two masking textures that I fed into the factors on mix shaders. The screen is

a screen capture from my actual phone.

The rest of the materials are very simple, either only one shader or a simple mix of two. The matte material which is the on the body has a noise texture as a bump map.



The multiply node is to control the strength of the bump

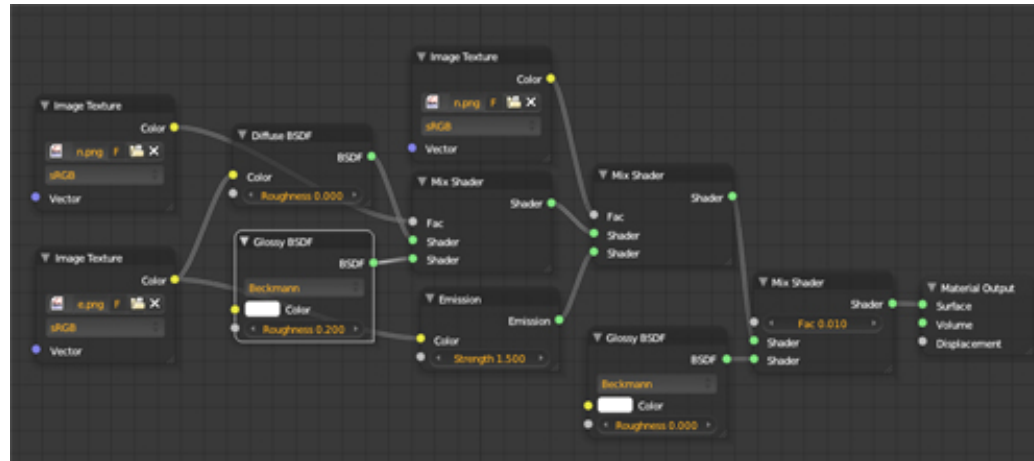
## Rendering and Compositing

I used Cycles for rendering. For lighting I made two tall mesh lights on both sides, equally strong and the strongest of them all, a big overhead mesh light and one slightly tilted behind the camera, mainly for reflection.

For compositing I used Photoshop. I rendered the phone from six angles and put it together into one image in Photoshop along with a slight color correction.



# MAKING OF : A Blade Phone



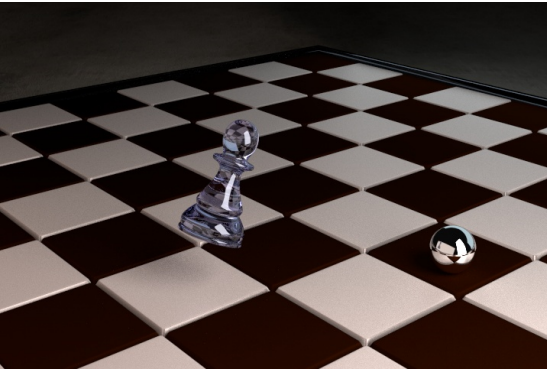
## In conclusion

I had a lot of fun making this image and learned a bunch in the process, I would do a lot of things differently if I would make it again.

You can see more of my work at <http://arnarg.netne.net/>

Happy blending and thanks for the read ●

# MAKING OF : A Pawn Still in a Hurry... 11 years later



by - **Dreamsgate**

## Introduction

Once upon a long time ago, okay about 11 years ago, there weren't many Blender tutorials to help new users. Luckily there was a tutorial guide, (Tutorial Guide #1) available from the Blender e-shop. Only way back then it wasn't the Blender Foundation, the main Blender site was still "Not a Number".

This tutorial guide was my introduction to how Blender worked and my favorite tutorial was "Pawn in a Hurry", written by Bart Veldhuizen (who now owns and runs Blender Nation). "Pawn in a Hurry" showed you how to do several cool things. You got to model a pawn using the "Spin" function, create a chess board and most importantly you got to animate the pawn springing across the chessboard.

When I finished the tutorial, I was amused to no end. What could be cooler than a pawn bouncing around? The Pawn even squashed and stretched during the bounce. Yes, I am very easily amused.

Eventually I finished the guide and continued on with my Blender learning, over time forgetting all about that wonderful little guide that introduced me to Blender. That is until recently. A couple of months ago, I was reminded of the "little Pawn" and just for giggles I pulled my now very battered guide off the shelf.

I was curious if the instructions still made sense considering all the changes in Blender since way back then. And surprise, they did. Well, as long as you are familiar the with basic functions in Blender it does.

So I set about recreating the pawn. I must admit I did do a few things differently. There is no longer a need to set "Spin" to 21, now we have subsurf, which allows a smooth model with far fewer Spin segments. I also added a few extra edge loops to define some of the edges. On the chessboard, I did add a very slight bevel, just to make it look nicer and catch highlights better. Other than that, I pretty much just followed along as it was originally written.

Once I finished setting the keyframes for the famous bounce, I sat back and watched it play. Guess what, it was just as amusing to see the pawn bounce realistically 11 years later. Yep, I am still just as easily amused. :P

Now granted this is still a basic tutorial, and I have learned a thing or two in the last 11 years so I decided to take it a little further. Instead of just one famous bounce, I bounced that poor little pawn all over the chessboard. And if you agree that one bounce is amusing, just wait until you have multiple bounces, I giggled myself silly.



And then since I had this nice little scene all set up, I decided to explore the Cycles render engine for a little realistic material rendering.

The tutorial took only an hour or so, even with the extra bounces, the exploration of Cycles Materials took a week or more. And for once, not because I was running into problems, I was just having too much fun playing with it to call it finished. :)

## MAKING OF : A Pawn Still in a Hurry... 11 years later



I can't even estimate how many material combinations I played with. And the best part was I could judge how well a material worked instantly. In fact watching the materials render was almost as fun as watching the Pawn bounce.

Besides the immediate feedback that Cycles offers, I was happy to discover that lighting seemed so much easier. At least for me. I experimented with several lighting setups using combinations of HDRI images and planes just to see how it affected my scene.

All said and done, I did end up with some nice materials and a lighting setup that looked reasonable, if somewhat uninspired. I rendered out the animation as PNGs, which sadly enough are still sitting in a folder waiting for me to run them through a few compositing nodes and the sequencer. I even have a nice "boing" sound file waiting to be used as well.

Obviously that is a project for another day. :)

In the meantime, it was fun revisiting my beginnings in Blender and proved to still be a good learning experience as I explored new tools that weren't available back then.

If you want to play with my little pawn, I have included the blend file in the blend zip download for this issue ●

# ARTICLE : Sketchfab



by - **Alban Denoyel**

## Introduction

Today, visual content is everywhere. Many websites offer great services to publish photos, videos, infographics, etc. But we couldn't find any good way to show 3D content online. Well, at least nothing that would meet our expectations: a web service to show 3D similar to your modeling software, technically advanced yet easy to use, with a nice user interface and the possibility to view other's creations.

Some people use 2D images, and even work on them to make them look better. Some use 360° photos. Other people use so-called "showreel" videos, others install plug-ins... But that's not enough. That's not interactive, that's not easy, and that's not user-friendly. Well, that's not 3D as we mean it in a connected world. And it's a pity when you agree that 3D creations are the richest computer graphic ones.

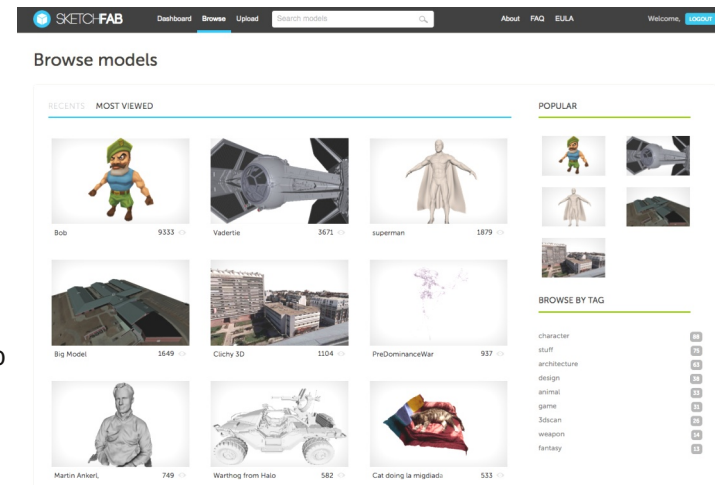
Do you remember the youtube revolution? Suddenly it was possible to get your video live in matter of seconds, publish it widely on the web,

share it, embed it... Now you can even edit it online. Well, that's what we want to do with Sketchfab.com, a cocoon for your 3D content.

On Sketchfab, it takes a minute to register and upload a model, and you get a decent viewer you can embed on your site, and a mini URL to share your creations. You can then have your own portfolio, and finally a professional and interactive way to show 3D content. You can also edit your biography to tell a bit more about yourself. On top of that, it's free!

Regarding technical specs, Sketchfab supports more than twenty 3d formats, including native blender files. It supports textures, lighting, and a lot more to come. We plan to add a social layer so that you can follow your favorite artists, comment and rate your favorite models. We will also add collaborative work features.

If you are a 3D artist looking for the best and easiest way to show your



work, come and join us on sketchfab.com

Some FAQs about SketchFab:

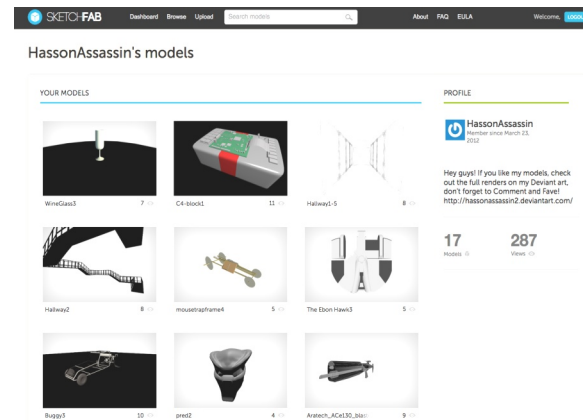
### 1. About Sketchfab

Sketchfab is a free service created in December 2011 which offers a simple way for 3D artists to upload and showcase their creations to the world. It doesn't require any application installation from the user to work, you just need a compatible browser.

### 2. Who is it for?

Sketchfab can be used by anyone who has 3D content: artists, designers, brands, architects, etc.

# MAKING OF : Sketchfab



## 3. Pricing?

Today, Sketchfab is free. We will keep a free version, but are working hard to provide an advanced and paid for version. If you have any suggestion about new features, let us know at [support@sketchfab.com](mailto:support@sketchfab.com)

## 4. How does it work?

You upload a model of your choice, the server will process it, you can add meta information and share it on the web. You can remove the models you uploaded at anytime from your dashboard. If you have texture to upload, you can pack all your files in a .zip file.

## 5. Which formats are supported?

Blender (.blend), Collada (.dae), Wavefront (.obj), OpenSceneGraph (.osg, osgt, osgb, ive), 3DS (.3ds), Lightwave (.lwo | .lws), Polygon File Format (.ply), Virtual Reality Modeling Language (.vrl), Open Inventor (.iv), Shape (.shp), Standard Tessellation Language (.stl), Biovision Hierarchy (.bvh), Open Flight (.flt), ac3d (.ac), DirectX (.x), Designer Workbench (.dw), 3DC point cloud (.3dc), carbon graphics Inc (.geo), Generic Tagged Arrays (.gta)

## 6. Where can I check new features?

Have a look at the changelog and follow us on twitter

## 7. Which browsers are compatible?

Firefox, Chrome, Opera, Safari. And more to come!

## 8. It doesn't work, what can I do ?

Make sure you have your graphics driver and your browser up to date. If that does not help, try one of the different browsers listed above. If it

still does not work, contact support.

## 9. How can I reach you?

General inquiries:  
[support@sketchfab.com](mailto:support@sketchfab.com)

Technical inquiries: Cédric Pinson,  
[cedric@sketchfab.com](mailto:cedric@sketchfab.com)

Commercial and press inquiries: Alban Denoyel, [alban@sketchfab.com](mailto:alban@sketchfab.com)

Follow us on twitter @sketchfab ●



# Want to write for BlenderArt Magazine ?

## Here is how

### Step1. Choose what you want to write

- Tutorials explaining Blender features, 3dconcepts, techniques or articles based on the focused theme of the issue
- Reports on useful Blender events throughout the world.
- Cartoons related to blender world.

### Step2. Send submissions to [sandra@blenderart.org](mailto:sandra@blenderart.org).

- Send us a notification on what you want to write and we can follow up from there.

### Step3. Some guidelines you must follow

- Images should be properly cut and represent the text appropriately.
- Images should be provided seperately in a folder named (images, img or pictures).
- Images should be named/labeled likewise (image1 or img1 etc).
- Provide proper captions for images if and when needed.
- Image format prefered is PNG but good quality JPG can also do.
- You can submit inline images in documents like DOC or Openoffice ODT etc but make sure the images were properly names before importing them in docs.
- Images inside a PDF are a strict no, but a pdf document with images if provided to show how the author wants the formatting of doc will be appreciated.
- Make sure that screenshots are clear and readable and the renders should be at least 800px, but not
- Text should be in either ODT, DOC, TXT or HTML.

### Step4. Archive them using 7zip or RAR or less

preferably zip.

### Step5. Additional stuff that you can do

- Please include the following in your email:
- Name: This can be your full name or blenderartist avtar.
- Photograph: As PNG and maximum width of 256Px. (Only if submitting the article for the first time )
- About yourself: Max 25 words .
- Website: (optional)

Note: All the approved submissions can be placed in the final issue or subsequent issue if deemed fit. All submissions will be cropped/modified if necessary. For more details see the blenderart website.

BA takes no responsibility fo the material in any form and the submission will automatically mean that you have agreed to the blenderart terms and conditions for submission for more information please do read the disclaimer.

## Disclaimer

blenderart.org does not takes any responsibility both expressed or implied for the material and its nature or accuracy of the information which is published in this PDF magazine. All the materials presented in this PDF magazine have been produced with the expressed permission of their respective authors/owners. blenderart.org and the contributors disclaim all warranties, expressed or implied, including, but not limited to implied warranties of merchantability or fitness for a particular purpose. All images and materials present in this document are printed/re-printed with expressed permission from the authors and or writers. The contents responsibility lies completely with the contributing writer or the author of the article.

This PDF magazine is archived and available from the blenderart.org website. The blenderart magazine is made available under Creative Commons' Attribution-NoDerivs 2.5' license.

COPYRIGHT© 2005-2011 'BlenderArt Magazine', 'blenderart' and BlenderArt logo are copyright of Gaurav Nawani. 'Izzy' and 'Izzy logo' are copyright Sandra Gilbert. All products and company names featured in the publication are trademark or registered trade marks of their respective owners.









